Amendments to the Claims

This listing of claims will replace all prior versions, and listings of claims in the application.

Claims 1-7. (Cancelled)

8. (New) A superscalar microprocessor for processing instructions, the microprocessor comprising:

an instruction fetch unit configured to fetch instructions from an instruction store according to a sequential program order;

a branch prediction circuit configured to provide a branch bias signal indicating whether a conditional branch controlled by a conditional branch instruction is predicted to be taken or not taken;

an instruction buffer coupled to receive fetched instructions from the instruction fetch unit and configured to make a plurality of fetched instructions, including an instruction selected according to the branch bias signal, concurrently available for execution;

a plurality of functional units configured to execute instructions, thereby generating result data;

a register file including a plurality of entries configured to store data including result data generated by the plurality of functional units, wherein each of the plurality of entries is accessible by reference to a respective location in the register file;

a decoder circuit configured to concurrently identify execution resources for more than one of the plurality of available instructions in the instruction buffer, the identified execution resources for each of the available instructions including a functional unit capable of executing the instruction and a register file entry corresponding to a source of an operand for the instruction;

an issue control circuit coupled to the decoder circuit and configured to concurrently issue more than one of the instructions from the instruction buffer to the functional units for execution, based on availability of the execution resources identified by the decoder circuit and without regard to the sequential program order;

a plurality of data routing paths coupled between the plurality of functional units and the register file and configured to concurrently transfer result data from more than one of the plurality of functional units to the register file;

bypass control logic coupled to the plurality of data routing paths and configured to supply result data from a first one of the plurality of functional units as operand data for another one or more of the plurality of functional units via an alternate data path that bypasses the register file, wherein supplying result data via the alternate data path occurs concurrently with transferring result data to the register file; and

retirement control logic coupled to the register file and configured to concurrently retire a plurality of instructions according to the sequential program order.

9. (New) The microprocessor of claim 8, wherein:

the plurality of functional units includes an integer functional unit and a floating-point functional unit; and

the bypass control logic is further configured such that an integer result from the integer functional unit is transferred to the floating-point functional unit via the alternate data path.

10. (New) The microprocessor of claim 8, wherein:

the plurality of functional units includes an integer functional unit and a floating-point functional unit; and

the bypass control logic is further configured such that a floating-point result from the floating-point functional unit is transferred to the integer functional unit via the alternate data path.

11. (New) The microprocessor of claim 8, further comprising:

operand data routing paths coupled between the register file and the functional units and configured to concurrently transfer operand data to more than one of the functional units.

12. (New) The microprocessor of claim 11, wherein the operand data routing paths transfer operand data directly from the register file to the functional units.

13. (New) The microprocessor of claim 8, wherein the register file includes:

a temporary buffer having a first plurality of entries; and

a retired register array having a second plurality of entries,

wherein the retirement control logic is further configured such that when an instruction is retired, corresponding result data is transferred from the temporary buffer to the retired register array.

14. (New) A method for processing instructions in a superscalar microprocessor, the method comprising:

fetching instructions from an instruction store according to a sequential program order;

predicting whether a conditional branch controlled by a conditional branch instruction included in the fetched instructions is taken or not taken;

making a plurality of fetched instructions, including an instruction selected according to the prediction, concurrently available in an instruction buffer for execution;

concurrently identifying execution resources for more than one of the plurality of available instructions in the instruction buffer, the identified execution resources for each of the more than one of the plurality of available instructions including a functional unit capable of executing the instruction and a register file entry corresponding to a source of an operand for the instruction;

concurrently issuing more than one of the plurality of available instructions from the instruction buffer for execution by a plurality of functional units, based on availability

of the identified execution resources for each instruction and without regard to the sequential program order;

executing the issued instructions in the plurality of functional units, thereby generating result data;

transferring the result data from the functional units to a register file, the register file including a plurality of entries, wherein each of the plurality of entries is accessible by reference to a respective location in the register file;

concurrently with said act of transferring, distributing the result data from a first one of the plurality of functional units as operand data for another one or more of the plurality of functional units via a bypass data path that bypasses the register file; and

retiring instructions according to the sequential program order.

15. (New) The method of claim 14, wherein:

the plurality of functional units includes an integer functional unit and a floating point functional unit; and

the act of supplying the result data includes supplying result data from the integer functional unit to the floating point functional unit via the bypass data path.

16. (New) The method of claim 14, wherein:

the plurality of functional units includes an integer functional unit and a floating point functional unit; and

the act of supplying the result data includes supplying result data from the floating point functional unit to the integer functional unit via the bypass data path.

17. (New) The method of claim 14, further comprising:

concurrently transferring operand data from the register file to more than one of the functional units via a plurality of operand data routing paths.

18. (New) The method of claim 17, wherein the operand data routing paths transfer operand data directly from the register file to the functional units.

19. (New) The method of claim 14, wherein the register file includes:

a temporary buffer having a first plurality of entries; and

a retired register array having a second plurality of entries,

wherein the act of retiring an instruction includes transferring corresponding result data from the temporary buffer to the retired register array.